

# An Analytic Expression of Performance Rate, Fitness Value and Average Convergence Rate for a Class of Evolutionary Algorithms

Jun He

**Abstract**—An important theoretical question in evolutionary computation is how good solutions evolutionary algorithms can produce. This paper aims to provide an analytic analysis of solution quality of evolutionary algorithms in terms of the performance rate, which is defined by the difference between 1 and the approximation ratio of the best solution found in each generation. The performance rate can be represented by a function of time. With the help of matrix analysis, it is possible to obtain an exact expression of such a function. For the first time, an analytic expression for calculating the performance rate is presented in this paper for a class of evolutionary algorithms, that is, (1+1) strictly elitist evolution algorithms. Furthermore, analytic expressions for calculate the fitness value and the average convergence rate in each generation are also derived for this class of evolutionary algorithms. The approach is promising, and it can be extended to non-elitist or population-based algorithms too.

## I. INTRODUCTION

Evolutionary algorithms (EAs) have been widely used to find good solutions to hard optimization problems. Many experimental results claim that EAs can obtain good quality solutions quickly. Nevertheless, from the viewpoint of the *NP*-hard theory, no efficient algorithm exists for solving *NP*-hard combinatorial optimization problems at the present and possibly for ever. Therefore it is unlikely that EAs are efficient in solving hard combinatorial optimization problems too. Instead of searching the exact solution to hard optimization problems, it is more reasonable to expect that EAs could find some good approximation solutions efficiently.

In theory, it is necessary to answer the question of how good approximation solutions EAs can produce to hard combinatorial optimization problems. Current work focuses on checking whether EAs are approximation algorithms. The research has attracted a lot of interests in recently years. Various combinatorial optimization problems have been investigated, including the minimum vertex cover problem [1], [2], the partition problem [3], the set cover problems [4], the minimum label spanning tree problem [5], and many others.

This paper studies the approximation performance of EAs from a different viewpoint. It aims to estimate the performance rate of the best solution in each generation, but without considering whether the EA is an approximation algorithm or not. The problem in this paper is described as follows: Given an EA for maximizing a fitness function  $f(x)$ , let  $f_{\max}$  be the optimal fitness and  $f_t$  the fitness of the best solution in the  $t$ -th generation. The approximation ratio of the  $t$ -th generation

solutions is  $f_t/f_{\max}$ . The performance rate of the EA [6] is

$$PR_t = 1 - \frac{f_t}{f_{\max}}. \quad (1)$$

which is the difference between 1 and  $f_t/f_{\max}$ . The performance rate  $PR_t$  is a function of  $t$ . Our research question is to find an analytic form of this function.

The perfect answer is to obtain a function  $\beta(t)$  in a closed form such that  $PR_t = \beta(t)$ . A good answer is to find a lower or upper bound  $\beta(t)$  such that  $PR_t \geq \beta(t)$  or  $PR_t \leq \beta(t)$ . Although it is more difficult to obtain a function  $\beta(t)$  in a closed form, it is still worth seeking it because a perfect answer is always our favorite. Fortunately for (1+1) strictly elitist EAs, an analytic expression of the performance rate has been constructed in this paper using matrix analysis. To the best of our knowledge, this is the first result of exactly expressing how the performance rate (also the fitness value and the convergence rate) changes as  $t$  on a class of EAs.

The paper is arranged as follows: Section II reviews the links to related work. Section III defines the performance rate. Section IV introduces Markov modeling. Section V makes a theoretical analysis. Section VI conducts a case study. Section VIII summarizes the paper.

## II. LINKS TO RELATED WORK

The performance rate can be regarded as a special case of the convergence rate. The convergence rate problem of an EA [7]–[9] asks the question of how fast  $f_t$  approaches  $f_{\max}$ ? A formal description is given as follows [9]. Suppose that the EA is modeled by a finite Markov chain with a transition matrix  $\mathbf{P}$ , in which a state is a population. Let  $\mathbf{p}_t$  be the probability distribution of the  $t$ -th generation population on a population space,  $\pi$  an invariant probability distribution of  $\mathbf{P}$ . Then  $\mathbf{p}_t$  is called convergent to  $\pi$  if  $\lim_{t \rightarrow \infty} \|\mathbf{p}_t - \pi\| = 0$  where  $\|\cdot\|$  is a norm. The convergence rate problem of an EA is to obtain a function  $\beta(t)$  such that  $\|\mathbf{p}_t - \pi\| = \beta(t)$ , or an upper bound  $\|\mathbf{p}_t - \pi\| \leq \beta(t)$  or a lower bound  $\|\mathbf{p}_t - \pi\| \geq \beta(t)$ . If let

$$\|\mathbf{p}_t - \pi\| = \frac{f_{\max} - f_t}{f_{\max}}, \quad (2)$$

then the convergence rate problem becomes the performance rate problem.

According to [10], there are two approaches to analyze the convergence rate of EAs for discrete optimization. The first

approach is based on the eigenvalues of the transition matrix associated with an EA. Suzuki [7] derived a lower bound of convergence rate for simple genetic algorithms through analyzing eigenvalues of the transition matrix. Schmitt and Rothlauf [11] found that the convergence rate is determined by the second largest eigenvalue of the transition matrix. The approach used in the current paper is the same as that in [7], [11]. All are based on analyzing the powers and eigenvalues of the transition matrix. The other approach is based on Doebelin's condition [9], [12]. Using the minorization condition in Markov chain theory, He and Kang [9] proved that for the EAs with time-invariant genetic operators, the convergence rate can be bounded by  $\epsilon^t$  where  $\epsilon \in (0, 1)$ .

The research in this paper is also linked to fixed budget analysis [13]. The purpose of fixed budget analysis is to find lower and upper bounds  $\beta_{low}(t)$  and  $\beta_{up}(t)$  such that  $\beta_{low}(t) \leq f_t \leq \beta_{up}(t)$  within fixed generations  $t$ . Usually assume that  $t$  is no more than a threshold. Jansen and Zarges [13] discussed two algorithms: random local search and the (1+1) EA and obtained such bounds on some well-known example problems. Fixed budget analysis focuses on bounding  $f_t$  itself for a fixed  $t$ ; but the performance rate focuses on the difference  $(f_t - f_{\max})/f_{\max}$ , which is within the framework of EAs' convergence rate. This makes the analysis in this paper completely different from [13].

### III. PERFORMANCE RATE

This section defines the performance rate of (1+1) EAs. Although the background of this paper is a special class of EAs, the concept of the performance rate can be applied to other types of EAs.

Consider a maximization problem, that is,  $\max\{f(x); x \in \mathcal{S}\}$  where  $\mathcal{S}$  is a finite set,  $f(x) \geq 0$  and  $f_{\max} > 0$ . An EA for solving the above problem is regarded as an iterative procedure: initially construct a population of solutions  $\Phi_0$ ; then given the  $t$ -th generation population  $\Phi_t$ , generate a new population  $\Phi_{t+1}$  in a probabilistic way. This procedure is repeated until an optimal solution is found. This paper investigate a class of (1+1) elitist EAs which are described in Algorithm 1. This kind of EAs is very popular in the theoretical analysis of EAs.

---

#### Algorithm 1 A (1+1) Strictly Elitist EA

---

```

1: initialize an individual  $\Phi_0$  and set  $t \leftarrow 0$ ;
2: while  $\Phi_t$  is not an optimal solution do
3:   generate a new individual  $\Psi_t$ ;
4:   if  $f(\Psi_t) > f(\Phi_t)$  then
5:     let  $\Phi_{t+1} \leftarrow \Psi_t$ ;
6:   else
7:     let  $\Phi_{t+1} \leftarrow \Phi_t$ ;
8:   end if
9:    $t \leftarrow t + 1$ ;
10: end while

```

---

The set  $\mathcal{S}$  is the state space of individuals. Given any  $t$ , the expression  $\Phi_t = X$  means that  $\Phi_t$  is a random variable

representing the  $t$ -th generation individual and  $X \in \mathcal{S}$  is its value. Given an initial individual  $\Phi_0 = X$ , the fitness of the  $t$ -th generation population  $\Phi_t$  is denoted by  $f(\Phi_t | \Phi_0 = X)$ . For the sake of notation,  $f(\Phi_t | \Phi_0 = X)$  is denoted by  $f(\Phi_t)$  in short. Since  $f(\Phi_t)$  is a random variable, we consider its expected value  $f_t \stackrel{\text{def}}{=} \mathbb{E}[f(\Phi_t)]$ . The approximation ratio of the  $t$ -th generation individual is  $f_t/f_{\max}$ . The approximation ratio of the optimal solution is 1.

*Definition 1:* The performance rate of the  $t$ -th generation individual is defined by

$$PR_t = 1 - \frac{f_t}{f_{\max}}. \quad (3)$$

In [6],  $PR_t$  is called the performance ratio. In order to avoid confusion with the approximation ratio, it is renamed the performance rate. This is inspired from the convergence rate which represents a difference:  $f_{\max} - f_t$ . The performance rate also represents a difference:  $1 - f_t/f_{\max}$ .

There is a link between the performance rate and fixed budget analysis [13]. From the definition of the performance rate, we get that the fitness value in the  $t$ -th generation equals to

$$f_t = f_{\max}(1 - PR_t). \quad (4)$$

There is a link between the performance rate and average convergence rate [14]. From the definition of the average convergence rate of an EA for  $t$  generations,

$$ACR_t \stackrel{\text{def}}{=} 1 - \left( \frac{f_{\max} - f_t}{f_{\max} - f_0} \right)^{1/t}, \quad (5)$$

we get that the average convergence rate for  $t$  generations equals to

$$ACR_t = 1 - \left( \frac{PR_t}{PR_0} \right)^{1/t}. \quad (6)$$

### IV. MARKOV CHAIN MODELLING FOR (1+1) STRICTLY ELITIST EAS

This section introduces Markov chain modelling for (1+1) strictly elitist EAs. The analysis follows the Markov chain framework which could be found in [14], [15].

Genetic operators in EAs could be either time-invariant or time-variant [9], [16]. This paper only considers time-invariant operators. Such an EA can be modeled by a homogeneous Markov chain with transition probabilities

$$\Pr(X, Y) \stackrel{\text{def}}{=} \Pr(\Phi_{t+1} = Y | \Phi_t = X), X, Y \in \mathcal{S}.$$

Let  $\mathbf{P}$  denote the transition matrix with entries  $\Pr(X, Y)$ . The state space  $\mathcal{S}$  is decomposed into two parts:  $\mathcal{S}_{\max}$  the set of optimal states and  $\mathcal{S}_{\text{non}} = \mathcal{S} \setminus \mathcal{S}_{\max}$ .

According to the strictly elitist selection, transition probabilities satisfy

$$\Pr(X, Y) = \begin{cases} \geq 0, & \text{if } f(Y) > f(X), \\ \geq 0, & \text{if } Y = X, \\ = 0, & \text{otherwise.} \end{cases} \quad (7)$$

Since  $\mathcal{S}$  is a finite state space, let vector  $(X_0, X_1, \dots, X_L)$  denote all states in  $\mathcal{S}$  where  $X_0$  represents the set of optimal states. Without the loss of generality, sort them according to the descending order,

$$f_{\max} = f(X_0) > f(X_1) \geq \dots \geq f(X_L) = f_{\min}.$$

Then transition matrix  $\mathbf{P}$  can be expressed as a lower triangular matrix:

$$\mathbf{P} = \begin{pmatrix} p_{0,0} & 0 & 0 & \dots & 0 & 0 \\ p_{1,0} & p_{1,1} & 0 & \dots & 0 & 0 \\ p_{2,0} & p_{2,1} & p_{2,2} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{L,0} & p_{L,1} & p_{L,2} & \dots & p_{L,L-1} & p_{L,L} \end{pmatrix}. \quad (8)$$

Let  $\mathbf{P}$  denote the sub-matrix which represents probabilities transitions among non-optimal states. It is a  $L \times L$  matrix, given as follows:

$$\mathbf{Q} = \begin{pmatrix} p_{1,1} & 0 & 0 & \dots & 0 & 0 \\ p_{2,1} & p_{2,2} & 0 & \dots & 0 & 0 \\ p_{3,1} & p_{3,2} & p_{3,3} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{L,1} & p_{L,2} & p_{L,3} & \dots & p_{L,L-1} & p_{L,L} \end{pmatrix}. \quad (9)$$

Let  $p_t(X)$  denote the probability of  $\Phi_t$  at the state  $X$ ,  $p_t(X) \stackrel{\text{def}}{=} \Pr(\Phi_t = X)$ . The vector

$$\mathbf{p}_t^T = (p_t(X_0), p_t(X_1), p_t(X_2), \dots)^T$$

denotes the probability distribution of  $\Phi_t$  in  $\mathcal{S}$  and the vector

$$\mathbf{q}_t^T = (p_t(X_1), p_t(X_2), \dots)^T$$

denote the probability distribution of  $\Phi_t$  in the non-optimal set  $\mathcal{S}_{\text{non}}$ . Here notation  $\mathbf{v}$  is a column vector and  $\mathbf{v}^T$  the row column with the transpose operation (denoted by  $^T$ ).

For any  $t \geq 1$ , the probability  $p_t(Y)$  (where  $Y \in \mathcal{S}_{\text{non}}$ ) equals to

$$\begin{aligned} P(\Phi_t = Y) &= \sum_{X \in \mathcal{S}_{\text{non}}} \Pr(\Phi_t = Y \mid \Phi_t = X) \Pr(\Phi_{t-1} = X) \\ &= \sum_{X \in \mathcal{S}_{\text{non}}} p_{t-1}(X) P(X, Y). \end{aligned}$$

It can be represented by matrix iteration

$$\mathbf{q}_t^T = \mathbf{q}_{t-1}^T \mathbf{Q} = \mathbf{q}_0^T \mathbf{Q}^t. \quad (10)$$

The initial probability distribution satisfies  $\mathbf{q}_0 \geq \mathbf{0}$  where  $\mathbf{0} = (0, 0, \dots)^T$ .

The expected fitness value  $f_t$  equals to

$$f_t = \sum_{X \in \mathcal{S}} f(X) \Pr(\Phi_t = X). \quad (11)$$

Then it follows

$$PR_t = \frac{f_{\max} - f_t}{f_{\max}} = \frac{\sum_{X \in \mathcal{S}_{\text{non}}} (f(X) - f_{\max}) q_t(X)}{f_{\max}}. \quad (12)$$

Let the vector

$$\mathbf{f}' \stackrel{\text{def}}{=} (f_{\max} - f(X_0), f_{\max} - f(X_1), f_{\max} - f(X_2), \dots)^T$$

represent the difference between the optimal fitness value and the fitness at each state. Let the vector

$$\mathbf{f}'_{\text{non}} \stackrel{\text{def}}{=} (f_{\max} - f(X_1), f_{\max} - f(X_2), \dots)^T$$

represent the difference between the optimal fitness value and the fitness at each non-optimal state. Then (12) can be rewritten in a vector form

$$\begin{aligned} PR_t &= \frac{f_{\max} - f_t}{f_{\max}} = \frac{\mathbf{p}_t^T \cdot \mathbf{f}'}{f_{\max}} = \frac{\mathbf{q}_t^T \cdot \mathbf{f}'_{\text{non}}}{f_{\max}} \\ &= \frac{\mathbf{q}_0^T \mathbf{Q}^t \mathbf{f}'_{\text{non}}}{f_{\max}}. \end{aligned} \quad (13)$$

From formula (13), we see that  $PR_t$  is determined by the initial distribution  $\mathbf{q}_0$ , the power of matrix  $\mathbf{Q}^t$ , the fitness difference  $\mathbf{f}'_{\text{non}}$  and the optimal fitness value  $f_{\max}$ . Only  $\mathbf{Q}^t$  changes as  $t$ , so it plays the most important role in determining the performance rate.

## V. AN ANALYTIC EXPRESSION OF PERFORMANCE RATE

This section gives an analytic expression of the performance rate for (1+1) strictly elitist EAs. The analysis is based on a result in matrix analysis on triangular matrices [17], [18].

From (13), we see that calculating  $PR_t$  becomes a mathematical problem of expressing the power matrix  $\mathbf{Q}^t$  once the initial probability distribution  $\mathbf{q}_0$  and the fitness difference  $\mathbf{f}'_{\text{non}}$  are known. For (1+1) strictly elitist EAs, the matrix  $\mathbf{Q}$  is a lower triangular matrix, and then it is feasible to calculate its power in a closed form. We follow the analysis in [17] and express matrix  $\mathbf{Q}^t$  explicitly in terms of its entries.

Since the analysis in [17] is for upper triangular matrices and  $\mathbf{Q}$  is lower triangular matrix, we consider the transpose of  $\mathbf{Q}$ . Let  $\mathbf{R} = \mathbf{Q}^T$  be the transpose of  $\mathbf{Q}$ . Denote  $\mathbf{R} = [r_{i,j}]$  with  $r_{i,j} = p_{i,j}$  for  $i, j = 1, \dots, L$ . For the sake of simplicity, assume that  $\mathbf{R}$  is a  $L \times L$  upper triangular matrix with unique diagonal elements. If the diagonal elements are not unique, a similar discussion can be conducted based on the result in [17, Theorem 1]. This will be given in a separate paper.

*Definition 2:* The power factors of  $\mathbf{R}$ ,  $[p_{i,j,k}]$  (where  $i, j, k = 1, \dots, L$ ), are recursively defined as follows:

$$p_{j,j,j} = r_{j,j}, \quad (14)$$

$$p_{i,j,k} = 0, \quad k < i \text{ or } k > j, \quad (15)$$

$$p_{i,j,k} = \frac{\sum_{l=k}^{j-1} p_{i,l,k} r_{l,j}}{r_{k,k} - r_{j,j}}, \quad i \leq k < j, \quad (16)$$

$$p_{i,j,j} = r_{i,j} - \sum_{l=i}^{j-1} p_{i,j,l}, \quad i < j. \quad (17)$$

*Lemma 1 (Lemma 1.2 in [17]):* Let  $\mathbf{R} = [r_{i,j}]$  be a non-singular upper triangular matrix with unique diagonal elements. Denote the entries of the matrix power  $\mathbf{R}^t$  by  $[r_{i,j}|t]$ . For any  $t \geq 1$ , if  $r_{i,j}|t = \sum_{k=i}^j p_{i,j,k} (r_{k,k})^{t-1}$ , then  $r_{i,j}|t+1 = \sum_{k=i}^j p_{i,j,k} (r_{k,k})^t$ .

*Proof:* Since  $\mathbf{R}^{t+1} = \mathbf{R}^t \cdot \mathbf{R}$ ,  $r_{i,l}|t = 0$  if  $l < i$  (because  $\mathbf{R}^t$  is upper triangular) and  $r_{l,j} = 0$  if  $l > j$  (because  $\mathbf{R}^t$  is

upper triangular), we have

$$r_{i,j|t+1} = \sum_{l=i}^j r_{i,l|t} r_{l,j}. \quad (18)$$

From the assumption:  $r_{i,j|t} = \sum_{k=i}^j p_{i,j,k}(r_{k,k})^{t-1}$ , and noting that  $p_{i,l,k} = 0$  if  $k > l$ , we have

$$\begin{aligned} r_{i,j|t+1} &= \sum_{l=i}^j r_{l,j} \sum_{k=i}^l p_{i,l,k}(r_{k,k})^{t-1} \\ &= \sum_{k=i}^j (r_{k,k})^{t-1} \sum_{l=k}^j r_{l,j} p_{i,l,k}. \end{aligned} \quad (19)$$

Notice that

$$\sum_{l=k}^j r_{l,j} p_{i,l,k} = \sum_{l=k}^{j-1} r_{l,j} p_{i,l,k} + r_{j,j} p_{i,j,k}. \quad (20)$$

Then substituting the sum in (20) by (16) in Definition 2, we have

$$\begin{aligned} \sum_{l=k}^j r_{l,j} p_{i,l,k} &= p_{i,j,k}(r_{k,k} - r_{j,j}) + r_{j,j} p_{i,j,k} \\ &= p_{i,j,k} r_{k,k}. \end{aligned} \quad (21)$$

Finally (19) is simplified as  $r_{i,j|t+1} = \sum_{k=i}^j p_{i,j,k}(r_{k,k})^t$ . This is the required conclusion. ■

**Lemma 2 (Theorem 1.3 in [17]):** Let  $\mathbf{R} = [r_{i,j}]$  be a nonsingular upper triangular matrix with unique diagonal elements. For any  $t \geq 0$ ,  $r_{i,j|t}$  denotes the  $(i,j)$ -th element of the matrix power  $\mathbf{R}^t$ , then

$$r_{i,j|t+1} = \sum_{k=i}^j p_{i,j,k}(r_{k,k})^{t-1} = \sum_{k=i}^j p_{i,j,k}(p_{k,k})^t. \quad (22)$$

*Proof:* According to (14), (15) and (17) in Definition 2, we see that (22) is true for  $t = 1$ . Then by induction, (22) is true for all  $t > 1$  from Lemma 1. ■

The above theorem gives an analytic expression of the matrix power  $\mathbf{R}^t$  and also  $\mathbf{Q}^t$ . Given a  $L \times L$  matrix  $\mathbf{Q}$ , the time complexity of calculating  $\mathbf{Q}^t$  is  $2^{\binom{L+2}{3}} + L(t-3)$  in terms of the number of multiplication and divisions according to [17].

Define coefficients

$$\alpha_k \stackrel{\text{def}}{=} \frac{\sum_{i=1}^L \sum_{j=i}^L f'_i p_{i,j,k} q_j}{f_{\max}}, \quad k = 1, \dots, L, \quad (23)$$

where  $\alpha_k$  is independent of  $t$ .

**Theorem 1:** Denote  $\mathbf{q}_0 = (q_1, \dots, q_L)^T$  where  $q_i = \Pr(\Phi_0 = X_i)$  is the initial probability of  $\Phi_0$  at state  $X_i$ . Denote  $\mathbf{f}'_{\text{non}} = (f'_1, \dots, f'_L)$  where  $f'_i = f_{\max} - f(X_i)$  is the fitness difference between  $f_{\max}$  and  $f(X_i)$ . Then for any  $t \geq 1$ , the performance rate is expressed by

$$PR_t = \sum_{k=1}^L \alpha_k (p_{k,k})^{t-1}. \quad (24)$$

*Proof:* From (13), we know

$$PR_t = \frac{\mathbf{q}_0^T \mathbf{Q}^t \mathbf{f}'_{\text{non}}}{f_{\max}} = \frac{(\mathbf{f}'_{\text{non}})^T \mathbf{R}^t \mathbf{q}_0}{f_{\max}}. \quad (25)$$

Using (22), we get

$$(\mathbf{f}'_{\text{non}})^T \mathbf{R}^t \mathbf{q}_0 = \sum_{i=1}^L \sum_{j=1}^L \sum_{k=i}^j f'_i p_{i,j,k} (p_{k,k})^{t-1} q_j. \quad (26)$$

According to Definition 2,  $p_{i,j,k} = 0$  if  $k < i$  or  $k > j$  and  $p_{i,j,k} = 0$  if  $i > j$ , then

$$\begin{aligned} (\mathbf{f}'_{\text{non}})^T \mathbf{R}^t \mathbf{q}_0 &= \sum_{i=1}^L \sum_{j=1}^L \sum_{k=1}^L f'_i p_{i,j,k} (p_{k,k})^{t-1} q_j \\ &= \sum_{k=1}^L (p_{k,k})^{t-1} \sum_{i=1}^L \sum_{j=i}^L f'_i p_{i,j,k} q_j. \end{aligned} \quad (27)$$

Using  $\alpha_k$ , (25) is rewritten as

$$PR_t = \sum_{k=1}^L \alpha_k (p_{k,k})^{t-1}. \quad (28)$$

The conclusion then is proven. ■

This theorem shows the performance rate is represented as a linear combination of exponential functions  $(p_{k,k})^t$  (where  $k = 1, \dots, L$ ). Here  $p_{k,k} = \Pr(\Phi_t = X_k \mid \Phi_{t-1} = X_k)$  is the probability of staying at the same state  $X_k$  in two generations. Other transition probabilities  $p_{i,j}, i \neq j$  only make contributions to coefficients  $\alpha_k$  (where  $k = 1, \dots, L$ ).

From the relationship between  $f_t$  and  $PR_t$  and that between  $ACR_t$  and  $PR_t$ , we get the following corollaries.

**Corollary 1:** The fitness in the  $t$ -th generation (where  $t \geq 1$ ) equals to

$$f_t = f_{\max} (1 - \sum_{k=1}^L \alpha_k (p_{k,k})^{t-1}). \quad (29)$$

**Corollary 2:** The average convergence rate for  $t$  generations (where  $t \geq 1$ ) equals to

$$ACR_t = 1 - \left( \sum_{k=1}^L \alpha_k (p_{k,k})^{t-1} \frac{f_{\max}}{f_{\max} - f_0} \right)^{1/t}. \quad (30)$$

In practice, the performance rate is calculated as follows:

- 1: given an initial probability distribution  $\mathbf{p}_0$ , the fitness difference  $\mathbf{f}'$  and matrix  $\mathbf{Q}$ ;
- 2: calculate power factors  $[p_{i,j,k}]$  where  $i, j, k = 1, \dots, L$  using Definition 2;
- 3: calculate coefficients  $[\alpha_k]$  (where  $k = 1, \dots, L$ ) using (23);
- 4: calculate the performance rate  $PR_t$  using Theorem 1.

## VI. CASE STUDY

This section provides a case study to illustrate the application of Theorem 1. The simple example is for the purpose of illustration. Nevertheless the theoretical result can be applied to any (1+1) strictly elitist EA on any fitness function if the corresponding transition matrix is with unique diagonal elements.

Consider a (1+1) elitist EA which adopts onebit mutation and elitist selection.

**Onebit Mutation.** Given a binary string, chose one bit at random and then flip it.

**Elitist Selection.** Choose the best from the parent and child as the next parent.

The EA is applied for maximizing two functions. The first function is the OneMax function.

$$f_1(x) = |x|. \quad (31)$$

where  $x$  is a binary string and  $|x| = x_1 + \dots + x_n$ . According to the  $f$  value, the space  $\mathcal{S}$  is split into  $n + 1$  subsets:

$$\mathcal{S}_l = \{x; |x| = n - l\}, \quad l = 0, 1, \dots, n. \quad (32)$$

Denote  $p_{i,j} = \Pr(\Phi_t \in \mathcal{S}_j \mid \Phi_{t-1} \in \mathcal{S}_i)$ . Calculate transition probabilities, given by

$$p_{i,j} = \begin{cases} \frac{i}{n}, & \text{if } j = i - 1, \\ 1 - \frac{i}{n}, & \text{if } j = i, \\ 0, & \text{otherwise.} \end{cases} \quad (33)$$

According to Theorem 1, for any  $t \geq 1$ , the performance rate is expressed by

$$PR_t = \sum_{k=1}^n \alpha_k \left(1 - \frac{k}{n}\right)^{t-1}. \quad (34)$$

where coefficients  $\alpha_k$  are given by

$$\alpha_k = \frac{\sum_{i=1}^L \sum_{j=i}^L f'_i p_{i,j,k} q_j}{f_{\max}}, \quad k = 1, \dots, L,$$

The power factors  $[p_{i,j,k}]$  (where  $i, j, k = 1, \dots, n$ ) are recursively defined as follows:

$$p_{i,j,k} = \begin{cases} \frac{n-j}{n}, & i = j = k, \\ 0, & k < i \text{ or } k > j, \\ \frac{j p_{i,j-1,k}}{j-k}, & i \leq k < j, \\ \frac{j}{n} - \sum_{l=i}^{j-1} p_{i,j,l}, & j = k, \text{ and } i = j - 1. \\ -\sum_{l=i}^{j-1} p_{i,j,l}, & j = k, \text{ and } i < j - 1. \end{cases}$$

The fitness differences  $[f'_i]$  (where  $i = 1, \dots, L$ ) are given by

$$f'_i = i.$$

Consider the case  $n = 4$ . Matrix  $\mathbf{R}$  is given by

$$\begin{pmatrix} 0.750 & 0.000 & 0.000 & 0.000 \\ 0.500 & 0.500 & 0.000 & 0.000 \\ 0.000 & 0.750 & 0.250 & 0.000 \\ 0.000 & 0.000 & 1.000 & 0.000 \end{pmatrix} \quad (35)$$

The fitness difference vector is

$$\mathbf{f}' = (1, 2, 3, 4)^T.$$

Choose the initial probability distribution in the non-optimal set to be

$$\mathbf{q}_0 = (0, 0, 0, 1)^T.$$

Using Definition 2, we calculate matrix  $[p_{i,j,k}]$  which is given by

$$[p_{1,j,k}] = \begin{pmatrix} 0.750 & 1.500 & 2.250 & 3.000 \\ 0.000 & -1.000 & -3.000 & -6.000 \\ 0.000 & 0.000 & 0.750 & 3.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \end{pmatrix},$$

$$[p_{2,j,k}] = \begin{pmatrix} 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.500 & 1.500 & 3.000 \\ 0.000 & 0.000 & -0.750 & -3.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \end{pmatrix},$$

$$[p_{3,j,k}] = \begin{pmatrix} 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.250 & 1.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \end{pmatrix},$$

$$[p_{4,j,k}] = \begin{pmatrix} 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \end{pmatrix}.$$

Using (23), we calculate coefficients  $\alpha_k$  (where  $k = 1, \dots, 10$ ) which are given by

$$(0.750, 0.000, 0.000, 0.000).$$

Recall that transition probabilities  $p_{k,k}$  (where  $k = 1, \dots, 4$ ) are

$$(0.750, 0.500, 0.250, 0.000).$$

Using (24), we calculate the performance rate which is given by

$$PR_t = 0.75^t. \quad (36)$$

The above formula shows that  $PR_t$  is determined by  $p_{1,1}$ , but  $p_{2,2}$  and  $p_{3,3}$  make no contribution to  $PR_t$ .

Furthermore, using Corollary 1, we calculate the fitness in each generation, given by

$$f_t = 4(1 - 0.75^t). \quad (37)$$

And using Corollary 2, we calculate the average convergence rate, given by

$$ACR_t = 1 - (0.75^t \times \frac{4}{4})^{1/t} = 0.25. \quad (38)$$

The above average convergence rate means that  $f_{\max} - f_t$  decays exponential quickly by a factor  $1 - 0.25 = 0.75$ , that is  $f_{\max} - f_t = 0.75^t(f_{\max} - f_0)$ .

The performance rate changes as the initial probability distribution  $\mathbf{q}_0$ . Now we show this point by choosing another  $\mathbf{q}_0$  as follows:

$$\mathbf{q}_0 = (0, 1, 0, 0)^T.$$

Using (23), we calculate coefficients  $\alpha_k$  (where  $k = 1, \dots, 10$ ) which are given by

$$(0.375, 0.000, 0.000, 0.000, \dots)^T$$

Using (24), we calculate the performance rate, given by

$$PR_t = 0.375 \times 0.75^{t-1}. \quad (39)$$

Furthermore, using Corollary 1, we calculate the fitness in each generation, given by

$$f_t = 4(1 - 0.375 \times 0.75^{t-1}). \quad (40)$$

And using Corollary 2, we calculate the average convergence rate, given by

$$\begin{aligned} ACR_t &= 1 - (0.375 \times 0.75^{t-1} \times \frac{2}{4})^{1/t} \\ &= 1 - 0.75 \times 0.25^{1/t}. \end{aligned}$$

Next we consider the quadratic function,

$$f_2(x) = |x|^2. \quad (41)$$

Given the class of pseudo-Boolean functions whose optimum is unique at  $(1, \dots, 1)$ , the quadratic and OneMax functions are the easiest to the  $(1+1)$  EA according to the theory of easiest and hardest fitness functions [19]. Their running times are identical. There are infinite easiest functions in this class, such as  $\exp(|x|)$  and  $\log(|x| + 1)$ .

Consider the case  $n = 4$ . Matrices  $\mathbf{R}$  and  $[p_{i,j,k}]$  are the same as (35). The fitness difference vector is

$$\mathbf{f}' = (7, 12, 15, 16)^T.$$

Choose the initial probability distribution of  $\mathbf{q}_0$  to be

$$\mathbf{q}_0 = (0, 0, 0, 1)^T.$$

Using (23), we calculate coefficients  $\alpha_k$  (where  $k = 1, \dots, 10$ ) which are given by

$$(1.313, -0.375, 0.000, 0.000, \dots)^T$$

Using (24), we calculate the performance rate, given by

$$PR_t = 1.313 \times 0.75^{t-1} - 0.375 \times 0.5^{t-1}. \quad (42)$$

The above formula shows that  $PR_t$  is determined by both  $p_{1,1}$  and  $p_{2,2}$ , but  $p_{3,3}$  makes no contribution to  $PR_t$ . Furthermore, using Corollary 1, we calculate the fitness in each generation, given by

$$f_t = 16(1 - 1.313 \times 0.75^{t-1} + 0.375 \times 0.5^{t-1}). \quad (43)$$

And using Corollary 2, we calculate the average convergence rate, given by

$$\begin{aligned} ACR_t &= 1 - [(1.313 \times 0.75^{t-1} - 0.375 \times 0.5^{t-1}) \times \frac{4}{4}]^{1/t} \\ &= 1 - (1.313 \times 0.75^{t-1} - 0.375 \times 0.5^{t-1})^{1/t}. \end{aligned}$$

Now we choose another  $\mathbf{q}_0$  as follows:

$$\mathbf{q}_0 = (0, 1, 0, 0)^T.$$

Using (23), we calculate coefficients  $\alpha_k$  (where  $k = 1, \dots, 10$ ) which are given by

$$(0.656, -0.063, 0.000, 0.000, \dots)^T$$

Using (24), we calculate the performance rate, given by

$$PR_t = 0.656 \times 0.75^{t-1} - 0.063 \times 0.5^{t-1}. \quad (44)$$

Furthermore, using Corollary 1, we calculate the fitness in each generation, given by

$$f_t = 16(1 - 0.656 \times 0.75^{t-1} + 0.063 \times 0.5^{t-1}). \quad (45)$$

And using Corollary 2, we calculate the average convergence rate, given by

$$\begin{aligned} ACR_t &= 1 - [(0.656 \times 0.75^{t-1} - 0.063 \times 0.5^{t-1}) \times \frac{2}{4}]^{1/t} \\ &= 1 - (0.328 \times 0.75^{t-1} - 0.0315 \times 0.5^{t-1})^{1/t}. \end{aligned}$$

Fig. 1 shows the performance rate  $PR_t$  of the EA on the OneMax and quadratic functions, which converges to 0. This rate clearly shows the solution quality in each generation in terms of the approximation ratio  $(= 1 - PR_t)$ .

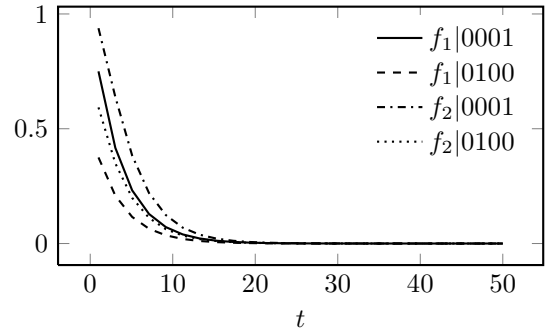


Fig. 1. Performance rate  $PR_t$  on functions  $f_1$  and  $f_2$  with  $n = 4$ . Two initial probability distributions  $\mathbf{q}_0 = (0, 0, 0, 1)^T$  and  $(0, 1, 0, 0)^T$ .

Fig. 2 demonstrates the fitness value  $f_t$  of the EA on the two functions, which converges to  $f_{\max}$  ( $= 4$  for  $f_1$  and  $= 16$  for  $f_2$ ). However, it doesn't provide any information about the solution quality and the convergence rate.

Fig. 3 illustrates the average convergence rate  $ACR_t$  of the EA on the two functions with two kinds of initialization. It converges to 0.25 in all four cases. From Fig. 3, we see that

- For  $f_1$  with  $\mathbf{q}_0 = (0, 0, 0, 1)$ , the EA converges as fast as an exponential decay:  $f_{\max} - f_t = 0.75^t(f_{\max} - f_0)$  (where  $0.75 = 1 - ACR_t = 1 - 0.25$ ).
- For both  $f_1$  and  $f_2$  with  $\mathbf{q}_0 = (0, 1, 0, 0)$ , the EA converges faster than the exponential decay:  $0.75^t(f_{\max} - f_0)$ .
- For  $f_2$  with  $\mathbf{q}_0 = (0, 0, 0, 1)$ , the EA converges slower than the exponential decay:  $0.75^t(f_{\max} - f_0)$ .

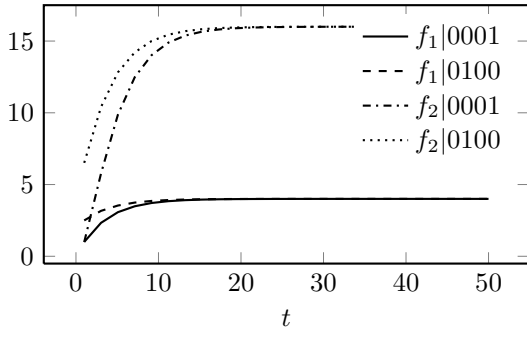


Fig. 2. Fitness value  $f_t$  on functions  $f_1$  and  $f_2$  with  $n = 4$ . Two initial probability distributions  $\mathbf{q}_0 = (0, 0, 0, 1)^T$  and  $(0, 1, 0, 0)^T$ .

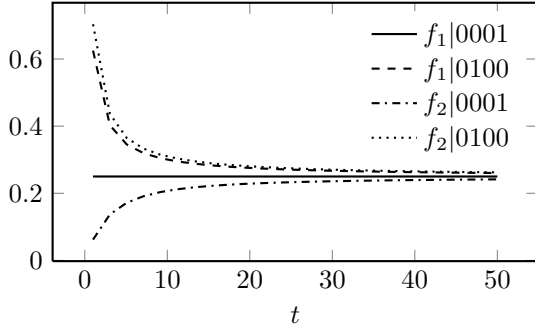


Fig. 3. Average convergence rate  $ACR_t$  on functions  $f_1$  and  $f_2$  with  $n = 4$ . Two initial probability distributions  $\mathbf{q}_0 = (0, 0, 0, 1)^T$  and  $(0, 1, 0, 0)^T$ .

Average convergence rate  $ACR_t$  converges to the spectral radius of  $\mathbf{Q}$  according to the theory of the average convergence rate [14, Theorem 1]. The average convergence rate can be used to compare how fast the same or different EAs converge to the optimum on the same or different problems.

At the end of this section, Table I lists the performance rates of the EA on six easiest functions. The table shows the analytic expressions of the performance rate are significantly different although all these functions are unimodal and their transition matrices are the same. Their running times are the same according to [19, Theorem 1], however their average convergence rates are different.

## VII. EXTENSION: NON-ELITIST OR POPULATION-BASED EAs

This section devotes to an extension of the analysis from (1+1) strictly elitist EAs to non-elitist or population-based EAs.

Assume that an EA is modelled by a Markov chain but its transition matrix  $\mathbf{P}$  (and then  $\mathbf{Q}$ ) is not a lower triangular matrix. According to Schur's triangularization theorem (in textbook [20, p508]), there exists an upper triangular matrix  $\mathbf{R}$  and unitary matrix  $\mathbf{U}$  such that  $\mathbf{Q}^T = \mathbf{U}\mathbf{R}\mathbf{U}^*$ . Then the matrix iteration (10) can be rewritten as follows,

$$\mathbf{q}_t = (\mathbf{Q}^T)^t \mathbf{q}_0 = \mathbf{U}\mathbf{R}^t \mathbf{U}^* \mathbf{q}_0. \quad (46)$$

Then the performance rate equals to

$$PR_t = \frac{(\mathbf{f}'_{\text{non}})^T \mathbf{U} \mathbf{R}^t \mathbf{U}^* \mathbf{q}_0}{f_{\max}}. \quad (47)$$

Let  $(\hat{\mathbf{f}}'_{\text{non}})^T = (\mathbf{f}'_{\text{non}})^T \mathbf{U}$  and  $\hat{\mathbf{q}}_0 = \mathbf{U}^* \mathbf{q}_0$ , then the performance rate can be rewritten as follows,

$$PR_t = \frac{(\hat{\mathbf{f}}'_{\text{non}})^T \mathbf{R}^t \hat{\mathbf{q}}_0}{f_{\max}}. \quad (48)$$

Since  $\mathbf{R}$  is an upper triangular matrix, the analysis of  $PR_t$  becomes the problem of expressing the matrix power  $\mathbf{R}^t$ . If  $\mathbf{R}$  is an upper triangular matrix with unique diagonal elements, Theorem 1 can be applied directly. If  $\mathbf{R}$  is an upper triangular matrix with equal diagonal elements, a similar analysis can be conducted (but in a separate paper). Therefore, in theory it is feasible to apply the approach to non-elitist or population-based EAs too.

Furthermore, even if transition probabilities are given only with approximate values, it is still possible for apply the same approach to bounding the performance rate. This will be discussed in the future.

## VIII. CONCLUSIONS

In this paper, the performance rate is used to measure the solution quality of evolutionary algorithms, that is

$$PR_t = 1 - \frac{f_t}{f_{\max}}. \quad (49)$$

Then for any (1+1) strictly elitist evolutionary algorithms, an analytic expression for calculating its performance rate is presented assume that the transition matrix corresponding to the EA is a upper triangular matrix with unique diagonal elements. The formula is

$$PR_t = \sum_{k=1}^L \alpha_k (p_{k,k})^{t-1}. \quad (50)$$

where  $p_{k,k} = \Pr(\Phi_t = X_k \mid \Phi_{t-1} = X_k)$  and coefficients  $\alpha_k$  are determined by (23). Based on the above formula, the analytic expressions of calculating the fitness value  $f_t$  in each generation and the average convergence rate  $ACR_t$  are also derived. Thus this paper gets an analytic expression of  $f_t$ ,  $PR_t$  and  $ACR_t$ .

The approach is promising and it is a new development of the theory of EAs' convergence rate. In theory, it is feasible to make a similar analysis for any EA if it is modeled by a Markov chain using Schur's triangularization theorem.

Our next work is to present a closed form for (1+1) strictly elitist evolutionary algorithms whose transition matrix is a upper triangular matrix but diagonal elements are not unique.

*Acknowledgment:*

The work was supported by the EPSRC under Grant EP/I009809/1.

TABLE I  
PERFORMANCE RATE OF THE (1+1) EA ON SIX EASIEST FUNCTIONS WHERE  $n = 10$  AND  $\mathbf{q}_0 = (0, \dots, 0, 1)$ .

function $f$	performance rate $PR_t$
$\log( x  + 1)$	$0.358 \times 0.9^{t-1} + 0.151 \times 0.8^{t-1} + 0.083 \times 0.7^{t-1} + 0.050 \times 0.6^{t-1} + 0.031 \times 0.5^{t-1} + 0.019 \times 0.4^{t-1} + 0.011 \times 0.3^{t-1} + 0.006 \times 0.2^{t-1} + 0.002 \times 0.1^{t-1}$
$ x $	$0.9 \times 0.9^{t-1}$
$ x ^2$	$1.710 \times 0.9^{t-1} - 0.720 \times 0.8^{t-1}$
$ x ^2 + \log( x  + 1)$	$1.678 \times 0.9^{t-1} - 0.700 \times 0.8^{t-1} + 0.002 \times 0.7^{t-1} + 0.001 \times 0.6^{t-1} + 0.001 \times 0.5^{t-1}$
$ x ^2 \times \log( x  + 1)$	$2.000 \times 0.9^{t-1} - 1.110 \times 0.8^{t-1} + 0.083 \times 0.7^{t-1} + 0.017 \times 0.6^{t-1} + 0.005 \times 0.5^{t-1} + 0.002 \times 0.4^{t-1} + 0.001 \times 0.3^{t-1}$
$\exp( x )$	$5.689 \times 0.9^{t-1} - 14.385 \times 0.8^{t-1} + 21.217 \times 0.7^{t-1} - 20.117 \times 0.6^{t-1} + 12.717 \times 0.5^{t-1} - 5.359 \times 0.4^{t-1} + 1.452 \times 0.3^{t-1} - 0.229 \times 0.2^{t-1} + 0.016 \times 0.1^{t-1}$

## REFERENCES

- [1] P. S. Oliveto, J. He, and X. Yao, "Analysis of the (1+1)-EA for finding approximate solutions to vertex cover problems," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1006–1029, 2009.
- [2] T. Friedrich, J. He, N. Hebbinghaus, F. Neumann, and C. Witt, "Approximating covering problems by randomized search heuristics using multi-objective models," *Evolutionary Computation*, vol. 18, no. 4, pp. 617–633, 2010.
- [3] C. Witt, "Worst-case and average-case approximations by simple randomized search heuristics," in *Proceedings of the 22nd annual conference on Theoretical Aspects of Computer Science*. Springer-Verlag, 2005, pp. 44–56.
- [4] Y. Yu, X. Yao, and Z.-H. Zhou, "On the approximation ability of evolutionary optimization with application to minimum set cover," *Artificial Intelligence*, no. 180-181, pp. 20–33, 2012.
- [5] X. Lai, Y. Zhou, J. He, and J. Zhang, "Performance analysis of evolutionary algorithms for the minimum label spanning tree problem," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 6, pp. 860–872, 2014.
- [6] J. He and X. Yao, "An analysis of evolutionary algorithms for finding approximation solutions to hard optimisation problems," in *Proceedings of IEEE 2003 Congress on Evolutionary Computation*. IEEE Press, 2003, pp. 2004–2010.
- [7] J. Suzuki, "A Markov chain analysis on simple genetic algorithms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25, no. 4, pp. 655–659, 1995.
- [8] —, "A further result on the markov chain model of genetic algorithms and its application to a simulated annealing-like strategy," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 28, no. 1, pp. 95–102, 1998.
- [9] J. He and L. Kang, "On the convergence rate of genetic algorithms," *Theoretical Computer Science*, vol. 229, no. 1-2, pp. 23–39, 1999.
- [10] L. Ming, Y. Wang, and Y.-M. Cheung, "On convergence rate of a class of genetic algorithms," in *Proceedings of 2006 World Automation Congress*. IEEE, 2006, pp. 1–6.
- [11] F. Schmitt and F. Rothlauf, "On the importance of the second largest eigenvalue on the convergence rate of genetic algorithms," in *Proceedings of 2001 Genetic and Evolutionary Computation Conference*, H. Beyer, E. Cantu-Paz, D. Goldberg, Parmee, L. Spector, and D. Whitley, Eds. Morgan Kaufmann Publishers, 2001, pp. 559–564.
- [12] L. Ding and L. Kang, "Convergence rates for a class of evolutionary algorithms with elitist strategy," *Acta Mathematica Scientia*, vol. 21, no. 4, pp. 531–540, 2001.
- [13] T. Jansen and C. Zarges, "Performance analysis of randomised search heuristics operating with a fixed budget," *Theoretical Computer Science*, vol. 545, pp. 39–58, 2014.
- [14] J. He and G. Lin, "Average convergence rate of evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, 2015. [Online]. Available: <http://dx.doi.org/10.1109/TEVC.2015.2444793>
- [15] J. He and X. Yao, "Towards an analytic framework for analysing the computation time of evolutionary algorithms," *Artificial Intelligence*, vol. 145, no. 1-2, pp. 59–97, 2003.
- [16] G. Rudolph, "Finite Markov chain results in evolutionary computation: a tour d'horizon," *Fundamenta Informaticae*, vol. 35, no. 1, pp. 67–89, 1998.
- [17] W. Shur, "A simple closed form for triangular matrix powers," *Electronic Journal of Linear Algebra*, vol. 22, pp. 1000–1003, 2011.
- [18] C. Huang, "An efficient algorithm for computing powers of triangular matrices," in *Proceedings of the 1978 ACM Annual Conference-Volume 2*. ACM, 1978, pp. 954–957.
- [19] J. He, T. Chen, and X. Yao, "On the easiest and hardest fitness functions," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 2, pp. 295 – 305, 2015.
- [20] C. Meyer, *Matrix Analysis and Applied Linear Algebra*. SIAM, 2000.